TRANSMITTAL OF APPEAL BRIEF (Large Entity)

Docket No.
INTL-0252-US

In Re Application Of: **Christopher J. Lord; Karl O. Lillevold; Gim Deisher**

Serial No.	Filing Date	Examiner	Group Art Unit
09/448,679	November 24, 1999	Trang U. Tran	2614

Invention: **NOISY EDGE REMOVAL FOR VIDEO TRANSMISSION**

TO THE ASSISTANT COMMISSIONER FOR PATENTS:

Transmitted herewith in triplicate is the Appeal Brief in this application, with respect to the Notice of Appeal filed on **January 22, 2003.**

The fee for filing this Appeal Brief is: **$320.00**

☒ A check in the amount of the fee is enclosed.

☐ The Commissioner has already been authorized to charge fees in this application to a Deposit Account. A duplicate copy of this sheet is enclosed.

☒ The Commissioner is hereby authorized to charge any fees which may be required, or credit any overpayment to Deposit Account No. **20-1504**
A duplicate copy of this sheet is enclosed.

RECEIVED
2003 MAR 25 AM 10: 17
BOARD OF PATENT APPEALS
AND INTERFERENCES

Signature

Dated: **March 20, 2003**

Mark J. Rozman, Reg. No. 42,117
TROP, PRUNER & HU, P.C.
8554 Katy Freeway, Suite 100
Houston, Texas 77024

21906
PATENT TRADEMARK OFFICE

I certify that this document and fee is being deposited on 3/20/2003 with the U.S. Postal Service as first class mail under 37 C.F.R. 1.8 and is addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231.

Signature of Person Mailing Correspondence

Jennifer Juarez

Typed or Printed Name of Person Mailing Correspondence

cc:

#9
J. Douglas
4/28/03

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Christopher J. Lord; Karl O. Lillevold; Gim Deisher	§	Group Art Unit:	2614
Serial No.:	09/448,679	§	Examiner:	Trang U. Tran
Filed:	November 24, 1999	§	Atty. Dkt. No.:	ITL.0252US
For:	NOISY EDGE REMOVAL FOR VIDEO TRANSMISSION	§		

Board of Patent Appeals & Interferences
Commissioner for Patents
Washington, D.C. 20231

RECEIVED
2003 MAR 25 AM 10: 17
BOARD OF PATENT APPEALS AND INTERFERENCES

APPEAL BRIEF

Sir:

Applicants respectfully appeal from the final rejection mailed October 22, 2002.

I. REAL PARTY IN INTEREST

The real party in interest is the assignee Intel Corporation.

II. RELATED APPEALS AND INTERFERENCES

None.

III. STATUS OF THE CLAIMS

Claims 1-30 are pending in the application. Claims 1-5, 9-12, 15-20, 22, and 25-30 are rejected. Each rejection is appealed. Claims 6-8, 13-14, 21, 23 and 24 are objected to and remain pending in the application.

Date of Deposit: March 20, 2003
I hereby certify under 37 CFR 1.8(a) that this correspondence is being deposited with the United States Postal Service as first class mail with sufficient postage on the date indicated above and is addressed to the Board of Patent Appeals & Interferences, Commissioner for Patents, Washington, DC 20231.

Jennifer Juarez

IV. STATUS OF AMENDMENTS

No amendments are presently pending in the application. Claims 1-5, 9-12, 15-20, 22 and 25-30 are the subject of this appeal.

V. SUMMARY OF THE INVENTION

In accordance with one embodiment of the invention, noisy edges in video frames may be removed to achieve higher frame rates and better quality video. The noisy edge removal mechanism may be used with applications which employ digital video encoding of captured content. Examples include video conferencing, video phone, network streaming video, and others. By detecting the presence of noisy edges in a video frame, a noisy line may be removed and replaced by a clean neighboring line prior to encoding. Noise may be removed from the top edge, either side edge, or the bottom edge of the video frame 10 as needed.

Turning to Figure 1, a noisy edge removal mechanism 20 may filter noise from a video frame 10 prior to entering a video encoder 18. In one embodiment of the invention, the noisy edge removal mechanism 20 includes a noisy edge detector 14 and a noisy edge filter 16. Initially, the noisy edge detector 14 receives a frame 10. The frame 10 is one of a plurality of frames 10 which make up a stream of video. The noisy edge removal mechanism 20 may be invoked for each frame 10 of the video stream, one at a time. For each frame 10, the noisy edge detector 14 analyzes one or more edges of the video frame 10. In one embodiment of the invention, an edge of the video frame 10 is selected, then divided into four equally sized portions. In Figure 2a, the top edge of the video frame 10 is divided into portions 20a, 21a, 22a, and 23a. These portions may be rows of the video frame 10, for example. See Specification, pages 2-3.

In Figure 2b, the left edge of the video frame 10 is divided into portions 20b, 21b, 22b, and 23b. These portions may be columns of the video frame 10, for example. In Figure 2c, the bottom edge of the video frame 10 is divided into portions 20c, 21c, 22c, and 23c. In Figure 2d, the right edge of the video frame is divided into portions 20d, 21d, 22d, and 23d. Once the edge of the video frame 10 is divided into portions of equal size, the portions are then subdivided into units of equal size. In Figure 3, a part of the video frame 10 of Figure 2a is subdivided into a plurality of units 24.

Each unit 24 of the video frame 10 is associated with a value. For example, a video display may be subdivided into pixels. Each pixel commonly has a value associated with the pixel, which may be stored in video memory. Each unit 24 of Figure 3 may likewise be associated with a distinct value.

In one embodiment of the invention, the noisy edge detector 14 determines the presence of noise based, in part, on comparisons between the values of the units 24 of the video frame 10. If adjacent units 24 are not similar, for example, noise may sometimes be inferred. So, once the video frame 10 is divided into discrete units 24, each one of which is assigned a value, mathematical operations may be performed to analyze the video frame 10.

Comparisons between values of the units 24 may be made using mathematical operations. In one embodiment, the values of the units 24 in one portion are compared to the values of the units 24 in a second, adjacent portion. The results of these comparisons are added together, to arrive at a result which is representative of the relationship between the two portions. A second pair of portions is likewise analyzed, supplying a second result, and so on. These results are then compared, and analyzed against one or more threshold values. In one embodiment of the

invention, the threshold values may be adaptable to the type of noise or other criteria. <u>See</u> Specification, p. 4-5.

In Figure 4, an analysis of the video frame 10, according to one embodiment of the invention, commences with the selection of an edge of the video frame (top, right, left, or bottom), subdivision of the video frame 10 into portions of equal size, and further subdivision in to units 24 (block 70). Once the edge of the video frame 10 has been subdivided into units 24, a pair of threshold values, T1 and T2, may be calculated (block 72). The threshold values are used to determine whether a value associated with one portion 20, 21, 22 or 23 of the video frame 10 varies significantly from a value associated with a second portion 20, 21, 22, or 23 of the video frame 10.

In one embodiment of the invention, these threshold values are based upon two variables, α and β. The values for α and β may be determined by analyzing one or more video frames 10 in which noise is known to be present. The values for α and β may also be based upon the source of the noise. For example, noise which results from the improper handling of closed captioning signals may produce a certain, predictable type of noise, to which a particular α value may be assigned. Alternatively, certain types of video capture devices may produce noise along the edges of the video frame, and thus a particular α or β variable may be appropriate. The α variable is presumed larger than the β variable, so that both a "stronger" (or larger) threshold value and a "weaker" (or smaller) threshold value may be used to analyze the edge of the video frame 10.

In one embodiment of the invention, once the α and β variables are known, T1 and T2 may be calculated based upon the following formulas:

$$T_1 = (\#\ \text{units/portion}) \times \alpha$$

$$T_2 = (\text{\# units/portion}) \times \beta$$

where $\alpha > \beta$. Because $\alpha > \beta$, the threshold value T1 is greater than the threshold value T2. See Specification, p. 5-6.

Looking back at Figure 4, a comparison of units 24 along an edge of the video frame 10 is performed (block 74). The comparison may be performed in a number of ways. In one embodiment of the invention, all units 24 of one portion are subtracted from all units 24 of an adjacent portion, to arrive at a plurality of results, the absolute values of which are then added together. This is called the sum of absolute differences, or SAD. In one embodiment, SAD values provide a discrete measure for analysis of the portions 20 through 23 of the video frame 10, not just the units 24 contained therein.

Turning back to Figure 4, once the SAD values for the portions 20 through 23 are determined, these values may be analyzed as well (block 76). In one embodiment of the invention, the "adjacent" SAD values are subtracted from one another, to arrive at one or more difference values, Dn. In one embodiment of the invention, once the difference values, D_1 and D_2, are calculated, they may then be analyzed against the threshold values, T_1 and T_2.

In accordance with one embodiment of the present invention, the analysis includes four comparisons between the values, D_1, D_2, T_1, and T_2, as shown in Figure 5. In one comparison, if D_2 is greater than T_1, because T_1 is the larger threshold value, noise is presumed to be found (diamond 82). Accordingly, two outermost portions, portion 20 and portion 21 of the video frame 10 are replaced with a third portion, portion 22, which is closer in from the edge of the video frame 10 (block 90).

Next, D_2 is compared to T_2 (diamond 84). If D_2 is larger than T_2, then D_2 is in between the two threshold values, T_1 and T_2. If, D_2 is between the two threshold values, and noise was found in the previous frame (as denoted by NOISEFOUND being TRUE), noise is presumed to

be found (diamond 84). The two outermost portions, portion 20 and portion 21, of the video frame 10 are replaced with a third portion, portion 22, which is closer in from the edge of the video frame 10 (block 90). <u>See</u> Specification, pp. 6-8.

In certain embodiments, calculations may identify noise by observing the spatial correlation between the portions 20 through 23 along the edge of the video frame 10. In other words, how similar portions 20 through 23 are to one another help to identify noise in the video frame 10. In addition to the calculations, noise detection in the previous frame may be included in analyzing the current frame. Once the noisy edge detector 14 has completed the analysis, the noisy edge filter 16 may replace one or more portions of the video frame 10 with a clean neighboring portion, in one embodiment of the invention. A new video frame 11 may then enter the video encoder 18.

A software program, for implementing one embodiment of the invention, shown in Figure 6, begins by clearing the Boolean variable, NOISEFOUND (block 102). NOISEFOUND indicates whether the previous frame required noise removal. An integer variable, FRAME, is also cleared to zero. FRAME keeps track of the current frame. FRAME is incremented (block 104). <u>See</u> Specification, p. 10.

For the current video frame received, the sum of absolute differences for the first four portions 20 through 23 of the video frame 10 is calculated (block 106). These calculations result in three values, SAD_{2021}, SAD_{2122}, and SAD_{2223}. Although four portions of the video frame 10 are analyzed in the example, this number may be adjusted to a larger or smaller number, as desired.

From the SAD values, two difference values, D_1, and D_2, are calculated. D_1 is the absolute value of the difference between SAD_{2021} and SAD_{2122}. Likewise, the second difference

value, D_2, represents the difference between SAD_{2122} and SAD_{2223}. The threshold values, T_1 and T_2, are calculated (block 110). Once the calculations D_1, D_2, T_1, and T_2 are completed, analysis of the video frame 10 for noise may begin.

In one embodiment of the invention, a series of queries determines whether the difference values D_1 and D_2 exceed the threshold values T_1 and T_2 (diamond 112). If the second difference value, D_2, is greater than the first threshold value, T_1, then noise has been detected. Accordingly, portions one and two of the video frame 10 are replaced with portion three (block 120). Further, the variable NOISEFOUND is set to TRUE (block 122), indicating that noise was found on the current frame. During analysis of subsequent frames, the variable NOISEFOUND is again tested. Next, if the second difference value, D_2, exceeds the second threshold value, T_2, and the variable NOISEFOUND is TRUE, then noise has again been detected (diamond 114). Again, portions one and two are replaced with portions three of the video frame 10 (block 120).

Where the first two calculations fail to result in noise detection, a second pair of inquiries may be initiated. The first difference value, D_1, is compared to the first threshold value, T_1 (diamond 116). If D_1 is larger, noise has been detected. In contrast to the result in block 120, however, only portion one is replaced with portion two (block 124). Otherwise, D_1 may be compared with the second threshold value, T_2. If D_1 is greater than T_2 and the variable NOISEFOUND is TRUE, then noise is detected (diamond 118). Again, portion one is replaced with portion two (block 124). The variable NOISEFOUND is set to TRUE (block 122). Otherwise, the variable NOISEFOUND is set to FALSE (block 126).

Following updates of the variable NOISEFOUND (block 122 and 126), the noisy edge removal mechanism 20 inquires whether the last frame has been reached (diamond 128). If so,

the operation is complete (block 130). Otherwise, the variable FRAME is incremented and the process is repeated (block 104). See Specification, pp. 10-12.

In Figure 7, in accordance with one embodiment of the invention, a processor-based system 70 may include a processor 30. The noisy edge removal mechanism 20 may be stored on the hard disk drive 44 such that, upon receiving the video input signal 62, the noisy edge removal program 20 is loaded into the memory 34 and executed. The video encoder 18, also stored on the hard disk drive 44 in one embodiment of the invention, may be used to encode the resulting frames. See Specification, pp. 12-13.

VI. ISSUES

A. Are Claims 1-5, 9, 15-20, and 22 Patentable Under 35 U.S.C. § 102(e) Over Acharya?

B. Are Claims 10 and 11-12 Patentable Under 35 U.S.C. § 102(e) Over Acharya?

C. Are Claims 25, 26 and 30 Patentable Under 35 U.S.C. § 102(e) Over Acharya?

D. Is Claim 27 Patentable Under 35 U.S.C. §102(e) Over Acharya?

E. Are Claim 28 and 29 Patentable Under 35 U.S.C. §102(e) over Acharya?

VII. GROUPING OF THE CLAIMS

For purposes of this appeal, Applicant has grouped together claims 1-5, 9, 15-20, and 22; claims 10-12; claims 25, 26 and 30; claim 27; and claims 28 and 29, as set forth above.

VIII. ARGUMENT

A. Claims 1-5, 9, 15-20, and 22 Are Patentable Under 35 U.S.C. § 102(e) Over Acharya

Claim 1 recites a method including receiving a video frame; identifying noise in a first portion of the video frame; and replacing the first portion with a second portion of the video frame. Claims 1-5, 9, 15-20 and 22 stand rejected under 35 U.S.C. § 102(e) over U.S. Patent No. 6,229,578 ("Acharya"). This rejection is improper. Nowhere does Acharya disclose (at least) "replacing the first portion with a second portion of the video frame" as recited by claim 1.

In an Advisory Action the Examiner incorrectly contends that Acharya discloses this element in col. 9, lns. 55-58 where Acharya states that "[a]lternatively, it may be desirable to use the dark current or reference pixels (of which there are usually several rows and columns) to substitute for missing values for edge pixels." Acharya, col. 9, lns. 55-58; Advisory Action dated January 13, 2003, page 2.[1] This portion of Acharya relates to the use of reference pixels for purposes of averaging with other pixels for obtaining an average pixel value (i.e., multi-level median hybrid filtering or a linear averaging). See Acharya, col.9, lns. 1-61. Nowhere does this or any other portion of Acharya disclose replacing a first portion of a video frame with a second portion of a video frame as recited by claim 1. Thus, for at least this reason, claim 1 and claims 2-5 depending therefrom are patentable. Similarly claim 9 and claim 15 depending therefrom, and claim 16 and claims 17-20 and 22 depending therefrom are also patentable.

Claim 1 is also patentable for the further independent reason that Acharya does not disclose a method in which "identifying noise in a first portion of the video frame" occurs, as recited by claim 1. In this regard, Acharya merely determines whether a pixel is an edge pixel or a non-edge pixel, and does not identify noise in a first portion of the video frame. In the Advisory Action, the Examiner contends (for the first time) that because Acharya discloses that

[1] It is telling that the Examiner recited this portion of Acharya for the very first time in the Advisory Action. Until then, the Examiner erroneously contended that replacement of a pixel by an average of neighboring pixels met the claimed "replacing the first portion with a second portion of the video frame." See Final Office Action dated October 22, 2002, p.2.

it is difficult to distinguish between noisy pixels and edge or detail pixels, Acharya meets this element. Advisory Action, p. 2. However, Acharya does not distinguish between noise and edges; it merely determines whether a pixel is an edge pixel or a non-edge pixel. E.g., Acharya, col. 5, lns. 11-18; col. 15, lns. 26-29. Thus the Examiner is incorrect in equating an edge pixel with noise. See Advisory Action, p.2. Thus, for this further reason, claim 1 and claims 2-5 depending therefrom are patentable. Similarly claim 9 and claim 15 depending therefrom, and claim 16 and claims 17-20 and 22 depending therefrom are also patentable.

B. Claims 10 and 11-12 Are Patentable Under 35 U.S.C. § 102(e) Over Acharya

Claim 10 depends from claim 9, and further recites that a software program that replaces a first portion of a video frame with a second portion of a video frame also writes to a memory to replace the first portion of the video frame. Claim 10 was also rejected under § 102(e) over Acharya. This rejection is improper.

As discussed above regarding claim 1 (see VIII.A), nowhere does Acharya disclose replacing a first portion of a video frame with a second portion of a video frame. More so, nowhere does Acharya disclose a "software program [that] writes to the memory to replace the first portion of the video frame." Instead, Acharya discloses that a resulting average value (i.e., not a second portion of the video frame) is stored into an array or table "so that the original value x (i, j) is not overwritten...." Acharya, col. 12, lns. 60-64. Accordingly, nowhere does Acharya disclose writing to memory to replace a first portion of a video frame. Instead, Acharya discloses the opposite. Thus for this further reason, claim 10 and claims 11-12 depending therefrom are patentable over Acharya.

C. Claims 25, 26 and 30 Are Patentable Under 35 U.S.C. § 102(e) Over Acharya

Claim 25 recites a method including receiving a video frame; analyzing a first portion of the video frame with a first adjacent portion of the video frame to obtain a first result; analyzing a second portion of the video frame with a second adjacent portion of the video frame to obtain a second result; and replacing the first portion of the video frame with one of the second portion, the first adjacent portion or the second adjacent portion if a comparison between the first result and the second result is indicative of noise.

Claims 25, 26 and 30 were also rejected under §102(e) over Acharya. This rejection is improper. Nowhere does Acharya disclose "replacing the first portion of the video frame with one of the second portion, the first adjacent portion or the second adjacent portion if a comparison between the first result and the second result is indicative of noise" as recited by claim 25. The Examiner contends that replacing a pixel under consideration with a linear average of the pixel and neighboring pixels as disclosed by Acharya meets this element. See Final Office Action, pp. 7-8. However, such an averaged pixel is not an original portion of a video frame. As such, the averaging technique of Acharya does not replace a first portion of a video frame with one of a second portion, a first adjacent portion, or a second adjacent portion, as recited by claim 25. Thus, claim 25 and dependent claims 26 and 30 are patentable over Acharya.

D. Claim 27 is Patentable Under 35 U.S.C. §102(e) Over Acharya

Claim 27 depends from claim 26 and further recites calculating a first threshold based upon an amount of plurality of units per the respective portion of the video frame. Claim 27 also stands rejected under §102(e) over Acharya.

Dependent claim 27 is patentable for the further reason that nowhere does Acharya disclose "calculating a first threshold based on an amount of the plurality of units per the

respective portion." In this regard, the Examiner contends that "dividing the gradient by the maximum gradient in the entire image" of Acharya meets this element. Advisory Action, p. 5. Clearly, this is erroneous as dividing one number (i.e., the gradient) by another number (i.e., the maximum gradient) bears no resemblance to basing a threshold upon an amount of the plurality of units per the respective portion. For this further reason claim 27 is patentable over Acharya.

E. Claims 28 and 29 Are Patentable Under 35 U.S.C. §102(e) over Acharya

Claim 28 depends from claim 27 and further recites that the first and second results "comprise a sum of absolute differences between the first portion and the first adjacent portion and the second portion and the second adjacent portion, respectively." The Examiner erroneously contends that Acharya meets this element by applying a mask to a pixel region to obtain a resulting differential. Advisory Action, p. 6 (citing Acharya, col. 6, lns. 10-29). However neither this nor any other portion of Acharya discloses that the first and second results "comprise a sum of absolute differences."

As disclosed in the present application, a sum of absolute differences takes "all units 24 of one portion [which are] all are subtracted from all units 24 of an adjacent portion to arrive at a plurality of results, the absolute values of which are then added together. This is called the sum of absolute differences, or SAD." Specification, p. 6, lns. 22-25. No such sum of absolute differences exists in Acharya, at least because the resulting differential of Acharya is not an absolute value. For this additional reason, claim 28 and claim 29 depending therefrom patentably distinguish over Acharya.

IX. CONCLUSION

Since the rejections of the claims are baseless, they should be reversed.

Respectfully submitted,

Date: March 20, 2003

Mark J. Rozman, Reg. No. 42,117
TROP, PRUNER & HU, P.C.
8554 Katy Fwy, Ste 100
Houston, TX 77024-1805
512/418-9944 [Phone]
713/468-8883 [Facsimile]

21906
PATENT TRADEMARK OFFICE

APPENDIX OF CLAIMS

The claims on appeal are:

1. A method comprising:
receiving a video frame;
identifying noise in a first portion of the video frame; and
replacing the first portion with a second portion of the video frame.

2. The method of claim 1, wherein identifying further comprises:
associating a noise level with the first portion of the video frame; and
comparing the noise level to a predetermined value.

3. The method of claim 2, wherein associating further comprises distinguishing the first portion from the second portion.

4. The method of claim 3, wherein distinguishing further comprises:
associating a first value with the first portion;
associating a second value with the second portion; and
performing a plurality of arithmetic operations between the first value and the second value.

5. The method of claim 4, wherein associating a first value with the first portion further comprises:
identifying a plurality of values associated with the first portion; and
performing an arithmetic operation on the plurality of values to render the first value.

9. A system including:
a bus;
a processor coupled to the bus;
a device coupled to the bus to receive a video signal; and
a storage medium coupled to the bus including a software program that, upon execution:

detects noise in a first portion of a video frame of the video signal; and

replaces a first portion of the video frame with a second portion of the video frame.

10. The system of claim 9, wherein the video frame is stored in a memory and, upon execution, the software program writes to the memory to replace the first portion of the video frame.

11. The system of claim 10, wherein, upon execution, the software program further detects noise by comparing a noise level associated with the first portion of the video frame with a predetermined value.

12. The system of claim 11, wherein the predetermined value is stored in the memory.

15. The system of claim 9, wherein the storage medium is a hard disk drive.

16. An article comprising a medium storing instructions that cause a processor-based system to:

locate a video frame of a video signal;

identify noise in a first portion of the video frame; and

replace the first portion with a second portion of the video frame.

17. The article of claim 16, further storing instructions that cause the processor-based system to locate the video frame by reading a memory device.

18. The article of claim 17, further storing instructions that cause the processor-based system to:

associate a noise level with the first portion of the video frame; and

compare the noise level to a predetermined value.

19. The article of claim 18, further storing instructions that cause the processor-based system to:

associate a first value with the first portion;

associate a second value with the second portion; and

perform a plurality of arithmetic operations between the first value and the second value.

20. The article of claim 19, further storing instructions that cause the processor-based system to:

identify a plurality of values associated with the first portion; and

perform an arithmetic operation on the plurality of values to render the first value.

22. The article of claim 16, wherein the medium storing instructions is a memory device.

25. A method comprising:

receiving a video frame;

analyzing a first portion of the video frame with a first adjacent portion of the video frame to obtain a first result;

analyzing a second portion of the video frame with a second adjacent portion of the video frame to obtain a second result; and

replacing the first portion of the video frame with one of the second portion, the first adjacent portion or the second adjacent portion if a comparison between the first result and the second result is indicative of noise.

26. The method of claim 25, wherein each of the first and second portions and the first and second adjacent portions comprises a plurality of units, and wherein the analyzing is performed on a unit by unit basis.

27. The method of claim 26, further comprising calculating a first threshold based upon an amount of the plurality of units per the respective portion.

28. The method of claim 27, wherein the first and second results comprise a sum of absolute differences between the first portion and the first adjacent portion and the second portion and the second adjacent portion, respectively.

29. The method of claim 27, wherein the comparison is indicative of noise if a difference between the first result and the second result exceeds the first threshold.

30. The method of claim 25, wherein the first portion comprises an edge portion of the video frame.